Coding Project

Problem Formulation & Specifications

This document presents the content and the specifications of the project that must be implemented in the context of the course "Coding Project" as well as the way in which the project must be delivered to the instructors of the course.

Recall: The evaluation of the course takes place only once per year and consists of testing the algorithmic know-how of the students as well as their coding skills. To this end, each student is requested to:

- Develop the mandatory project according to the modalities specified below.
- Carry out a written exam to check his/her theoretical knowledge and problem-solving skills.

Note that:

- **The project is mandatory, and no student may access the written exam if the whole work delivered is judged insufficient.**
- **The score of the project plus the sum of the scores relative to supplementary exercises that the lecturer may disseminate during the course account for at most 6 points of the final grade**.
- The written exam accounts for **at most 14 points** of the final grade.

Important notes: The project can be carried out by a group of maximum 3 persons. Note however that, because the examination is individual, the final note of a student may be different from the other students in the same group.

- Do not share your code with other colleagues: any two equal or very similar codes will be scored **0**.
- Any piece of code that does not compile or does not run as expected will be considered as not delivered and will be scored **0**.

Chaussée de Binche 151, M1.01.01 7000 Mons, Belgium Office: B-302. Tel: +32 (0)65 32 33 13. Email: daniele.catanzaro@uclouvain.be

Specifications 2021-2022

Please address the questions stated in the next pages and develop the appropriate Python codes to solve them.

- Colaboratory is the platform that the instructors will use to test and evaluate python code. Get acquainted with it as soon as possible. Once your project is complete, please send an email the **assistant and the instructor** of the course containing the link to the Colaboratory code, as well as the name, surname and matriculation number of the students that have authored the project. As email object use exclusively the following text: Coding Project 2021/2022. **Please ensure that your source code is available to the instructors of the course. Any inaccessible code will be scored 0.**
- The deadline for the delivery of your project is fixed to: December 27 2020 at 23:59. **Any email received after this deadline will be scored 0 as well as any change carried out on Colaboratory after the deadline.**
- The scheduling of the defenses will be uploaded about 1 week before the exam.
- Do not cheat. Any cheating attempt will be severely punished.

Problem formulation

Tesla Corporation wants to improve its travel planner that allows the drivers of its electric cars to define the trip from a city A to a city B. To this end, Tesla engaged us as consultants and provided us with an Excel File containing two sheets:

- CityInfo, including the names of several European cities as well as the state they belong to, their population, their latitude, and their distances (in km) from among all of the other cities.
- CountryGraph, including the adjacency matrix of the states that are considered in CityInfo.

The models that Tesla wants to consider are Tesla Model S and Tesla Model X. The former has an autonomy of 420 km, while the latter of 359 km.

1. Tesla wants to provide the user of the travel planner with the possibility to change model of car when planning a travel.
2. Moreover, Tesla wants that the travel planner may
    a. show the distance matrices per country
    b. show the cities within a given country
    c. show the shortest path between any two cities.

In particular, note that the shortest path might not be feasible for a given car, in the sense that it may include arcs having a distance superior to the car capacity. For example, the following path between Brussels and Barcelona is feasible:

Brussels => Paris => Limoges => Toulouse => Sabadell => Barcelona

Distance between consecutive cities:

Brussels	=> Paris	: 300.52 km.	Power reserve: 119.48 km
Paris	=> Limoges	: 392.22 km.	Power reserve: 27.78 km
Limoges	=> Toulouse	: 286.93 km.	Power reserve: 133.07 km
Toulouse	=> Sabadell	: 307.43 km.	Power reserve: 112.57 km
Sabadell	=> Barcelona	: 22.32 km.	Power reserve: 397.68 km

The path is feasible and will make you arrive with a power reserve of 397.68 km!

In contrast, the following path instead between Brussels and Seville is not feasible:

Brussels => Paris => Tours => Bordeaux => San Sebastian => Salamanca => Seville

Distance between consecutive cities:

Brussels	=> Paris	: 300.52 km.	Power reserve: 119.48 km
Paris	=> Tours	: 239.45 km.	Power reserve: 188.55 km
Tours	=> Bordeaux	: 346.18 km.	Power reserve: 73.82 km
Bordeaux	=> San Sebastian	: 236.13 km.	Power reserve: 183.87 km
San Sebastian	=> Salamanca	: 452.20 km.	Power reserve: -32.20 km
Salamanca	=> Seville	: 460.03 km.	Power reserve: -40.03 km

In fact, the path includes arcs having distances longer than the car autonomy. For example the ride San Sebastian => Salamanca (452.20 km) is longer than car capacity (420.00 km), assuming to travel at 130

km/h. Similarly, the ride Salamanca => Seville (460.03 km) is longer than car capacity (420.00 km), assuming to travel at 130 km/h.

The planner must give the possibility to input any origin city and any destination city from among those listed in the excel file. The planner must ensure a robust input management, i.e., anticipate possible input errors of the user when writing origin and destination (e.g., "Brusels" instead of "Brussels").

You may assume that during the trip, the driver fully recharges the battery of the car at each intermediate city between origin and destination. The Power Reserve is defined to be the difference between the CarCapacity and the distance between two cities. When the user inputs an origin city, the planner must immediately highlight in green the paths that are feasible, in blue those that are "critical" (i.e., arcs whose power reserve is <=25km) and in red those that are unfeasible (i.e., arcs whose power reserve is negative, hence beyond the car capacity). If a path is unfeasible, the planner should try to search for an alternative (if it exists) and highlight the difference in terms of km between the shortest path and the new feasible path found.

For example, a possible feasible alternative for the trip Brussels-Seville is

Brussels => Paris => Tours => Bordeaux => San Sebastian => Burgos => Madrid => Cordoba => Seville

Distance between consecutive cities:

Brussels	=> Paris	: 300.52 km.	Power reserve: 119.48 km
Paris	=> Tours	: 239.45 km.	Power reserve: 180.55 km
Tours	=> Bordeaux	: 346.18 km.	Power reserve: 73.82 km
Bordeaux	=> San Sebastian	: 236.13 km.	Power reserve: 183.87 km
San Sebastian	=> Burgos	: 213.39 km.	Power reserve: 206.61 km
Burgos	=> Madrid	: 241.35 km.	Power reserve: 178.65 km
Madrid	=> Cordoba	: 393.25 km.	Power reserve: 26.75 km
Cordoba	=> Seville	: 137.43 km.	Power reserve: 282.57 km

The path is feasible and will make you arrive with a power reserve of 282.57 km!
The new path is 73.18 km longer than the shortest path!

3. Tesla wants that the travel planner may classify all the shortest paths between any two pairs (origin, destination) in a given input excel file as feasible, critical and unfeasible.

   The output must be provided in the following form:
   Origin => Destination | Power reserve: xxx | Path ….

   As an example: Salzburg => Trondheim | Power reserve: -76.09 | Path: Salzburg => Regensburg => Berlin => Malmö => Göteborg => Oslo => Trondheim

4. The travel planner must be able to list, for each origin, all the cities that are reachable and those that are not reachable. Moreover, it must be able to show a table that summarizes how many cities can be reached from a given origin and how many cannot be reached. It must also be able to save this last information in an excel file.

5. The planner must give the user the possibility to navigate the possible choices of the user with a menu as shown by the lecturer during the presentation of the project.